Accelerators in a Hybrid HPC World: How Can Applications Benefit?

Introduction

For quite some time, so-called accelerators are considered to be core components of successful HPC systems. While the power of GPU integration paved the way, today's approaches based on FPGA, ASIC, and even Quantum Computing technology become more and more important. The result is a powerful but complex and hybrid HPC world where programming such systems is THE challenge. The de-facto HPC programming standards (OpenMP and MPI) used by computer scientists are not appropriate for life science researchers that prefer high-level support such as given by Python and R environments. And in the industry world we can still find demands for application-specific Java interfaces. Our approach bridges user needs from all three communities by providing tailor-made interface layers that share commonly needed system components.

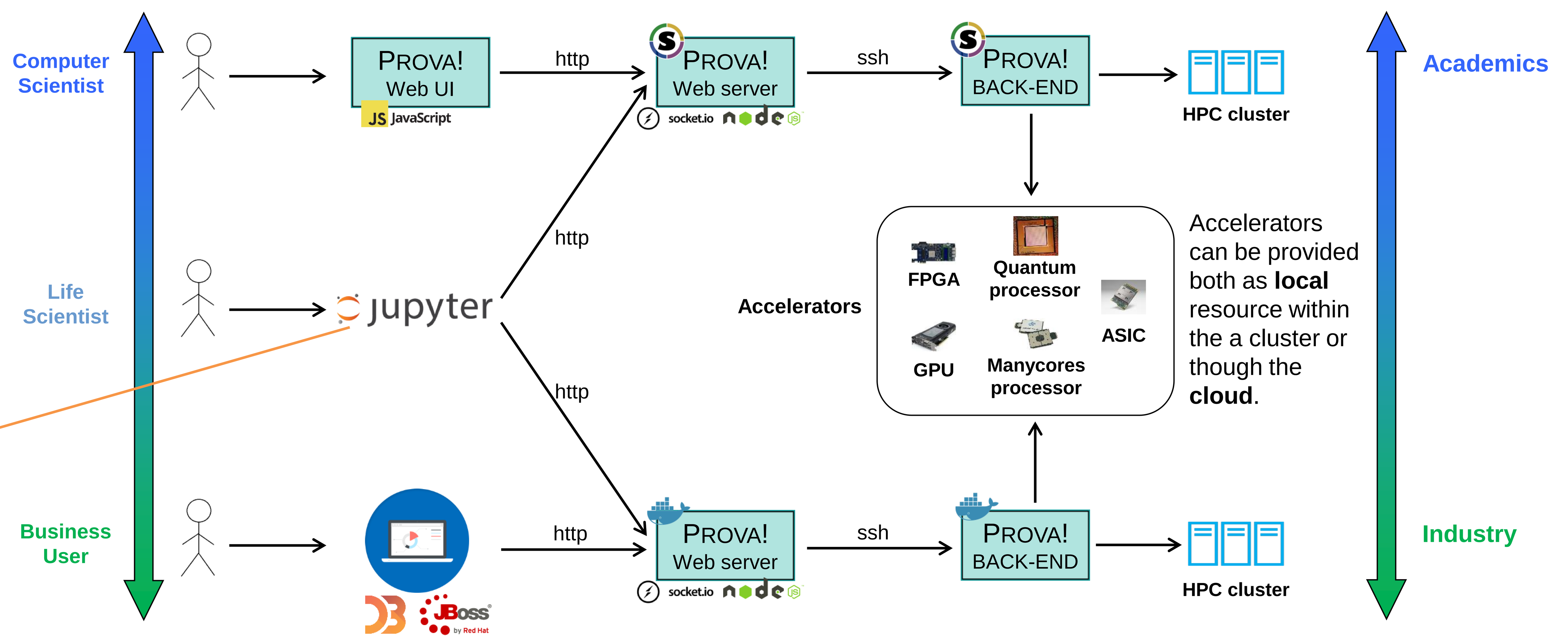
Objectives

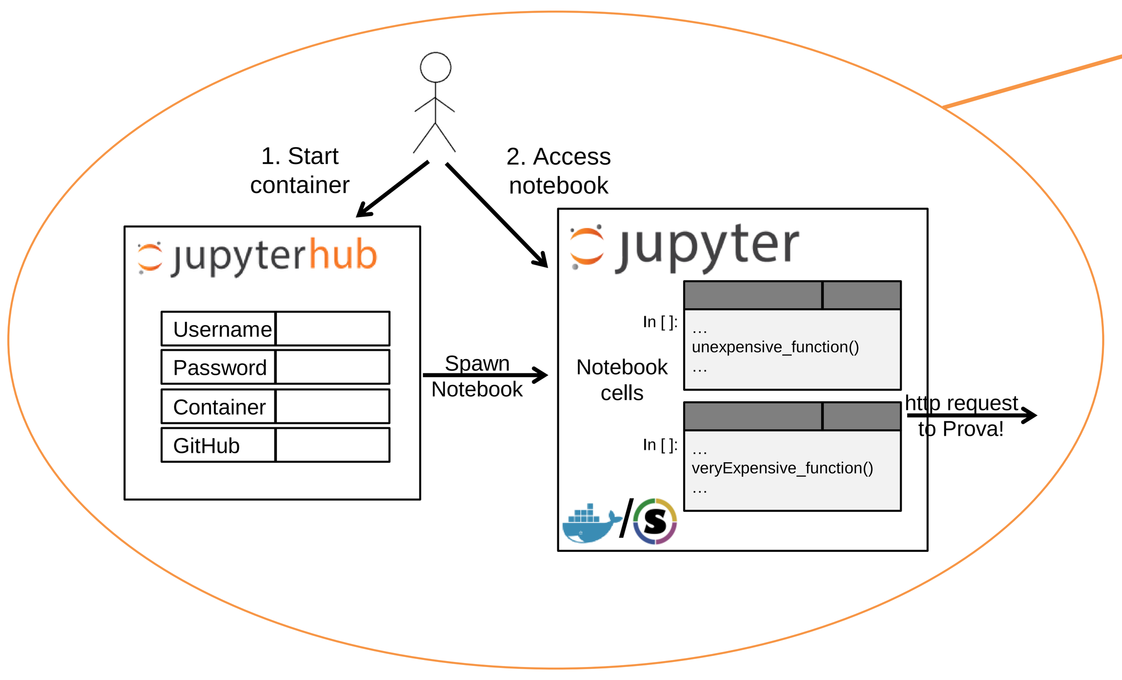
- Allow scientists to exploit the real computational capabilities of an HPC System in a **transparent** manner.
- Empower users from **different communities** to access HPC resources without changing the way they carry on their experiments.
- Achieve **reproducibility** proposing a systematic approach to the experiments.
- Using a **modular architecture** for better interfacing new systems and technologies.

Prototype System: High-Level Architectural View

The designed architecture is mainly organized in three parts: An **interface** (on the left) sending http requests to a **web server** (in the middle) that cares about the ssh connection to a **remote system** (on the right).

Re-using some of the components provided by PROVA! (web server and back-end), it is possible to implement different interfaces creating more execution workflows.





The Jupyter block in the architecture is composed of a JupyterHub (jupyterhub.readthedocs.io) and a Jupyter notebook: the scientist uses JupyterHub to spawn and manage notebooks which are based on a specific container image (Docker/Singularity). The PROVA! notebook extension allows the scientist to send requests to the web server for executing a notebook cell on a remote system when needed.

Applications

Progress (as of Feb '18)

- Benchmark of stencil compilers
  - Using PROVA! to reproduce and compare results obtained on different remote HPC systems within the university[1].

Next Steps

- Supervised machine learning for American Option Pricing[2]
  - Run different versions of the code on both local and remote HPC systems using the extended Jupyter notebook.
- Deep Neural Networks for Anomaly Detection in Lung Imaging[3]
  - Use the extended Jupyter Notebook to run the GPU implementation on resources available within the company or from the cloud and compare the performance.
- Stochastic Optimization for Supply Chain Management
  - Execute the code on a remote HPC system from a custom web application that provides the visualization of the results.

State of PhD work

Progress (as of Feb '18)

- Experiment execution and reproduction management system using a software build and installation framework (EasyBuild) and/or container solution (Docker/Singularity).
- Implementation of a client/server extension for Jupyter Notebook interfacing with the PROVA!.

Next Steps

- Use JupyterHub for spawning container-based Jupyter Notebooks and use the container image for running the same experiment of the remote system.
- Interface PROVA! Back-end with classical and emerging HPC resources provided as cloud service.
- Provide quantitative and qualitative analysis of system performances and user acceptance.

References

[1] Guerrera, D., Burkhart, H., Maffia, A. *Reproducible Stencil Compiler Benchmarks Using* PROVA!. In Proceedings of the 7th International Workshop on Performance Modeling, Benchmarking and Simulation of High Performance Computing Systems, 2016.

[2] Godina, T., Mu, G., Maffia, A., Sun, Y. (2017, September 14-15). *Supervised Machine Learning with Control Variate for American Option Pricing*. Paper presented at Code4Life Scientific: Data Science Algorithms, Techniques and Architectures in Healthcare, Poznań, Poland. (http://scientific.code4life.pl)

[3] Krzysztof, K., Szymon, T. (2017, September 14-15). *Deep Neural Networks for Anomaly Detection in Lung Imaging - Preliminary Result*. Paper presented at Code4Life Scientific: Data Science Algorithms, Techniques and Architectures in Healthcare, Poznań, Poland. (http://scientific.code4life.pl)